

Application No.: 10/698,820

Docket No.: MWS-062

AMENDMENTS TO THE DRAWINGS

The attached sheets of drawings include changes to Figure 4.

Attachment: Replacement sheet

MAY 22 2007

Application No.: 10/698,820

Docket No.: MWS-062

REMARKS

In this Response, claims 12 and 16-26 have been amended. Claims 1-26 are currently pending, of which claims 1, 12 and 16 are independent. No new matter has been added.

I. Objection to the Drawings

The Examiner objected to the drawings as failing to comply with 37 CFR 1.84(p)(5) because the drawings do not include reference numerals 85, 87, 89, 91 and 95 as mentioned on page 8 of the Specification (office action, paragraph 7). Accordingly, in the amendments to the drawings, Applicants have amended Figure 4 to include reference numerals 85, 87, 89, 91, 93 and 95. In view of the amendments to the drawings, Applicants respectfully request reconsideration and withdrawal of the objection to the drawings.

II. Objection to the Claims

The Examiner objected to claims 16-26 because they allegedly do not have proper explicit antecedent basis (office action, paragraph 8). In the claim amendments, Applicants have amended claim 16 so that it recites "said computer-readable medium," rather than "said medium." Accordingly, Applicants have also amended claims 17-26 so that they recite "the computer-readable medium," rather than "the medium." In view of the amendments to claims 16-26, Applicants respectfully request reconsideration and withdrawal of the objection to the claims.

III. Summary of Rejections

Claims 12-26 stand rejected under 35 U.S.C. §101.

Claims 1-4, 6-8, 12-14, 16-19 and 21-23 stand rejected under 35 U.S.C. §102(b) as being anticipated by United States Patent Publication Number 2002/0010908 to Cheng et al (hereafter "Cheng").

Claims 5, 15 and 20 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Cheng in view of United States Patent Publication Number US 2004/0085357 to Childress et al (hereafter "Childress").

Application No.: 10/698,820

Docket No.: MWS-062

Claims 9, 10, 24 and 25 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Cheng in view of United States Patent Publication Number US 2003/0225774 to Davidov et al (hereafter "Davidov").

Claim 11 and 26 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Cheng and in view of United States Patent Number US 6,066,181 to DeMaster (hereafter "DeMaster").

These rejections will be discussed separately below.

IV. Claim Rejections under 35 USC §101

The Examiner rejected claims 12-26 under 35 U.S.C. §101 as allegedly being directed to non-statutory subject matter (office action, paragraph 10). Applicants respectfully traverse the 35 U.S.C. §101 rejections of claim 12-26 as set forth below.

Regarding claims 12-15, the Examiner alleged that "Such claimed computer programs do not define any structural and functional interrelationships between the computer program and other claimed elements of a computer, which permit the computer program's functionality to be realized" (office action, paragraph 10). Applicants respectfully disagree. Independent claim 12 recites a tangible electronic device including computer program components. Applicants contend that claim 12 adequately defines structural and functional interrelationships between the computer program components and the electronic device which permits the computer program components' functionality to be realized. As such, claim 12 recites statutory subject matter.

Regarding claims 16-26, the Examiner alleged that "the limitation of "said [computer-readable] medium holding instructions" can be reasonably interpreted as the computer-readable medium carrying or transmitting electrical signals, since the instructions are not recorded on the computer-readable medium" (office action, paragraph 10). Applicants respectfully disagree. Independent claim 16 recites a computer-readable medium including software instructions. Applicants contend that claim 16 adequately defines structural and functional interrelationships between the software instructions and the computer-readable medium which permits the

Application No.: 10/698,820

Docket No.: MWS-062

software instructions' functionality to be realized. As such, claim 16 recites statutory subject matter. See *In re Beauregard*, 53 F.3d 1583, 1583-84 (Fed. Cir. 1995).

In view of the foregoing arguments, Applicants respectfully request reconsideration and withdrawal of the outstanding rejection of claims 12-26 under 35 U.S.C. §101.

V. Claim Rejections under 35 USC §102(b)

The Examiner rejected claims 1-4, 6-8, 12-14, 16-19 and 21-23 under 35 U.S.C. §102(b) as being anticipated by Cheng (office action, paragraph 12). Applicants respectfully traverse the 35 U.S.C. §102(b) rejections of claims 1-4, 6-8, 12-14, 16-19 and 21-23 as set forth below.

The Cheng reference discusses a code generation engine that automatically generates software code for a handler function using one of handler function information and parameter information (Cheng, abstract). Parameter information and handler function information is received via a graphical user interface and is used to automatically generate handler function code (Cheng, abstract).

A. Claim 1

Independent claim 1 recites:

"In an electronic device having a graphical modeling and execution environment, said graphical modeling and execution environment including at least one graphical model, a method comprising the steps of:

providing a user interface with a plurality of selectable parameters for a custom storage class, said *custom storage class specifying the manner in which an automatic code generator creates source code corresponding to data referenced by said graphical model in said graphical modeling and execution environment*; and

creating a custom storage class in said graphical modeling and execution environment utilizing parameters selected by a user from said plurality of selectable parameters." [emphasis added]

Applicants respectfully submit that the Cheng reference fails to disclose at least the following feature of independent claim 1: "*custom storage class specifying the manner in which*

Application No.: 10/698,820

Docket No.: MWS-062

an automatic code generator creates source code corresponding to data referenced by said graphical model in said graphical modeling and execution environment," because Cheng does not disclose a custom storage class.

The Examiner states at paragraph 12 of the office action:

"As per Claim 1, Cheng et al. disclose:

- providing a user interface with a plurality of selectable parameters for a custom storage class, said custom storage class specifying the manner in which an automatic code generator creates source code corresponding to data referenced by said graphical model in said graphical modeling and execution environment (see Figures 4, 6, and 7; Paragraph [0026], "FIG. 7 shows an exemplary GUI 400 for command node editor 120."); Paragraph [0028], "The entering of parameters is also accomplished via GUI 400 by adding the desired parameters to parameter field 410."; Paragraph [0043], "Handler code generation engine 135 automatically generates this software code using the information entered by the developer and the parameter and handler function definitions generated by command structure generation engine 145.");"

The Examiner further states in the remarks section at paragraph 17 of the office action:

"Note that the handler function definitions and parameter definitions are interpreted as "custom storage class," where handler code generation engine automatically generates software code using the information from the handler function definitions and parameter definitions."

It appears from the Examiner's remarks at paragraph 17 of the office action that the Examiner is pointing to the handler function definitions and parameter definitions in Cheng as disclosing "a custom storage class," as required by claim 1. As recited in claim 1, a custom storage class specifies the *manner in which an automatic code generator creates source code corresponding to data referenced by a graphical model*. Additionally, as described in Applicants' Specification at page 2, lines 5-8, changes to the unique set of instructions defining a custom storage class collectively apply to the (potentially large) set of data of that class.

Cheng does not disclose a custom storage class as required by Applicants' claim 1, because the parameter definitions and handler function definitions in Cheng do not specify the manner in which an automatic code generator creates source code corresponding to data

Application No.: 10/698,820

Docket No.: MWS-062

referenced by a graphical model. Cheng states: "Command structure generation engine 145 takes the information entered by the developer and generates the handler function definition file. The exemplary code above may include the following information: the type of command (e.g., can this command handle "No" forms), the bitmask of required parameters, the bitmask of optional parameters and the actual handler function associated with the definition" (Cheng, paragraph [0042]). As described in the above excerpt of Cheng, the parameter definitions and handler function definitions generated by the command structure generation engine correspond to *parameter functions and handler functions*, respectively. In contrast, claim 1 requires generation of source code corresponding to *data referenced by a graphical model*. Thus, Cheng fails to disclose a custom storage class as recited in claim 1 and as described in Applicants' Specification.

Applicants also respectfully submit that Cheng fails to disclose "custom storage class specifying the manner in which an automatic code generator creates source code corresponding to data referenced by said *graphical model* in said graphical modeling and execution environment," as recited in claim 1.

Regarding the above feature of claim 1, the Examiner further states in the remarks section at paragraph 17 of the office action:

"c) Examiner disagrees. Cheng et al. clearly disclose that the software code associated with the handler function "corresponds to data referenced by a graphical model," as required by claim 1 (see Figure 4; Paragraph [0023], 'FIG. 4 shows an exemplary command graphical user interface ("GUF") 200 for command structure manifest 110 described with respect to FIG. 2. Command structure manifest 110 enables a developer to visually manipulate the command structure by adding and deleting command nodes at any level.' And "... GUT (sic) 200 also shows parameters and handler functions associated with each command node."). Note that the handler function code is generated as a result of a new command node being inserted into the command structure."

It appears from the Examiner's remarks at paragraph 17 of the office action that the Examiner is pointing to handler function code generated via GUI 200 as disclosing "source code corresponding to data referenced by said *graphical model*," as required by claim 1. Applicants respectfully disagree because Cheng does not disclose creation of source code corresponding to

Application No.: 10/698,820

Docket No.: MWS-062

a *graphical model*. Graphical user interface (GUI) 200 in Cheng, pointed out by the Examiner, is not a graphical model. In Cheng, GUI 200 is used merely to present command structure manifest 110 to a developer (Cheng, paragraph [0023] and Figure 4). In contrast, the graphical model recited in claim 1 is described to be executable in Applicants' Specification at page 1, lines 24-27, and is not simply a GUI for the presentation of information. Cheng does not disclose that GUI 200 is a graphical model, and thus GUI 200 in Cheng is not synonymous with a graphical model as required by claim 1.

In view of the above arguments, Applicants respectfully request reconsideration and allowance of claim 1.

B. Claims 2-4 and 6-8

Claims 2-4 and 6-8 depend from independent claim 1 and, as such, incorporate all of the elements of claim 1. Accordingly, claims 2-4 and 6-8 are allowable for at least the reasons set forth above with respect to claim 1. Applicants therefore respectfully request reconsideration and allowance of claims 2-4 and 6-8.

C. Claim 12

Amended independent claim 12 recites:

"An electronic device having a modeling and execution environment with at least one graphical model, said electronic device comprising:
a user interface with a plurality of selectable parameters for a custom storage class, said *custom storage class specifying the manner in which an automatic code generator creates source code from said graphical model*;
a custom storage class in said graphical modeling and execution environment, said custom storage class created utilizing parameters selected by a user from said plurality of selectable parameters; and
a view of salient aspects of the source code generated by said automatic code generator utilizing the user-selected parameters." [emphasis added]

Applicants respectfully submit that Cheng fails to disclose at least the following feature of claim 12: "a user interface with a plurality of selectable parameters for a custom storage class, said custom storage class specifying the manner in which an automatic code generator creates

Application No.: 10/698,820

Docket No.: MWS-062

source code from said graphical model." As discussed above in connection with claim 1, Cheng fails to disclose a custom storage class, and therefore Cheng does not support a valid 35 U.S.C. §102(b) rejection of claim 12. Applicants respectfully request reconsideration and allowance of claim 12.

D. Claims 13 and 14

Claims 13 and 14 depend from independent claim 12 and, as such, incorporate all of the elements of claim 12. Accordingly, claims 13 and 14 are allowable for at least the reasons set forth above with respect to claim 12. Applicants therefore respectfully request reconsideration and allowance of claims 13 and 14.

E. Claim 16

Amended independent claim 16 recites:

"A computer-readable medium for use in an electronic device having a graphical modeling and execution environment, said graphical modeling and execution environment including at least one graphical model, said computer-readable medium holding instructions for:

providing a user interface with a plurality of selectable parameters for a custom storage class, said *custom storage class specifying the manner in which an automatic code generator creates source code corresponding to data referenced by said graphical model in said graphical modeling and execution environment*; and

creating a custom storage class in said graphical modeling and execution environment utilizing parameters selected by a user from said plurality of selectable parameters." [emphasis added]

Applicants respectfully submit that Cheng fails to disclose at least the following feature of claim 16: "providing a user interface with a plurality of selectable parameters for a custom storage class, said custom storage class specifying the manner in which an automatic code generator creates source code corresponding to data referenced by said graphical model in said graphical modeling and execution environment." As discussed above in connection with claim 1, Cheng fails to disclose a custom storage class, and therefore Cheng does not support a valid

Application No.: 10/698,820

Docket No.: MWS-062

35 U.S.C. §102(b) rejection of claim 16. Applicants respectfully request reconsideration and allowance of claim 16.

F. Claims 17-19 and 21-23

Claims 17-19 and 21-23 depend from independent claim 16 and, as such, incorporate all of the elements of claim 16. Accordingly, 17-19 and 21-23 are allowable for at least the reasons set forth above with respect to claim 16. Applicants therefore respectfully request reconsideration and allowance of claims 17-19 and 21-23.

VI. Claim Rejections under 35 USC §103(a)

A. Claims 5, 15 and 20

The Examiner rejected claims 5, 15 and 20 under 35 U.S.C. §103(a) as being unpatentable over Cheng in view of Childress (office action, paragraph 14). Applicants respectfully traverse the 35 U.S.C. §103(a) rejections of claims 5, 15 and 20 as set forth below.

The Cheng reference has been summarized above.

The Childress reference discusses creating, viewing and/or modifying business rules used by an automated insurance claim processing system (Childress, paragraphs [0009-0010]). A rule editor may provide a user with a graphical display of at least a portion of a business rule implemented in software (Childress, paragraphs [0009-0010]). Business rules of knowledge-based system encode the formulas used in evaluating insurance claims in an insurance claim processing software (Childress, paragraph [0007]).

I. Claims 5 and 20

Claims 5 and 20 depend from independent claims 1 and 16, respectively. Applicants respectfully submit that Cheng and Childress, alone or in any reasonable combination, fail to disclose or suggest all of the features of claims 5 and 20.

Application No.: 10/698,820

Docket No.: MWS-062

Applicants respectfully submit that Cheng and Childress, alone or in any reasonable combination, fail to disclose or suggest at least the following feature of claims 5 and 20: "providing a user interface with a plurality of selectable parameters for a custom storage class, said custom storage class specifying the manner in which an automatic code generator creates source code corresponding to data referenced by said graphical model in said graphical modeling and execution environment."

As discussed previously, Cheng fails to disclose or suggest the above feature of claims 1 or 16 from which claims 5 and 20, respectively, depend. The teachings of Childress do not supplement Cheng in such a way as to cure Cheng's failure to disclose or suggest the above feature of claims 5 and 20.

For example, Childress fails to disclose or suggest a custom storage class, as required by claims 5 and 20. Childress discusses generating code corresponding to *business rule components in the form of entries in a database table*. In contrast, claims 5 and 20 both require generating source code corresponding to *data referenced by a graphical model*. A database table as discussed in Childress is not synonymous with a graphical model as recited in claims 5 and 20. Thus, Childress also fails to disclose or suggest generating source code corresponding to data referenced by a graphical model, as recited in claims 5 and 20.

For at least the reasons presented above, Cheng and Childress, alone or in any reasonable combination, fail to disclose or suggest the features of claims 5 and 20. Therefore, Cheng and Childress do not support a valid 35 U.S.C. §103(a) rejection of claims 5 and 20.

ii) Claim 15

Claim 15 depends from independent claim 12. Applicants respectfully submit that Cheng and Childress, alone or in any reasonable combination, fail to disclose or suggest all of the features of claim 15.

Applicants respectfully submit that Cheng and Childress, alone or in any reasonable combination, fail to disclose or suggest at least the following feature of claim 15: "a user interface with a plurality of selectable parameters for a custom storage class, said custom storage

Application No.: 10/698,820

Docket No.: MWS-062

class specifying the manner in which an automatic code generator creates source code from said graphical model."

As discussed previously, Cheng fails to disclose or suggest the above feature of claim 12 from which claim 15 depends. The teachings of Childress do not supplement Cheng in such a way as to cure Cheng's failure to disclose or suggest the above feature of claim 15.

For example, Childress fails to disclose or suggest a custom storage class, as required by claim 15. Childress discusses generating code from *business rule components in a database table*. In contrast, claim 15 requires generating source code from a *graphical model*. Business rule components in a database table as discussed in Childress are not synonymous with a graphical model as recited in claim 15. Thus, Childress also fails to disclose or suggest generating source code from a graphical model, as recited in claim 15.

For at least the reasons presented above, Cheng and Childress, alone or in any reasonable combination, fail to disclose or suggest the features of claim 15. Therefore, Cheng and Childress do not support a valid 35 U.S.C. §103(a) rejection of claim 15.

B. Claims 9, 10, 24 and 25

The Examiner rejected claims 9, 10, 24, and 25 under 35 U.S.C. §103(a) as being unpatentable over Cheng in view of Davidov (office action, paragraph 15). Applicants respectfully traverse the 35 U.S.C. §103(a) rejections of claims 9, 10, 24 and 25 as set forth below.

The Cheng reference has been summarized above.

The Davidov reference discusses an infrastructure for creating applications for mobile information devices, using a tag-based markup language (Davidov, paragraph [0013]). Developers can use the markup language to define applications and content based on easily manipulated textual tags, rather than having to write specific code (Davidov, paragraph [0013]). A compiler, which is optionally controllable via a command line interface, processes the tags in several phases (Davidov, paragraph [0013]). Initially, the input is parsed in order to check for errors (Davidov, paragraph [0013]). Next, a hierarchical object model of the application is

Application No.: 10/698,820

Docket No.: MWS-062

populated with objects corresponding to the tags (Davidov, paragraph [0013]). Then, source code files are generated, which include supporting resource files (such as images) corresponding to the objects in the hierarchy (Davidov, paragraph [0013]). In producing the source code files, a generator traverses the object model hierarchy in a top-down manner, producing a source file for different elements, for example, screens, forms, and servlets (Davidov, paragraph [0013]). For each element required to be generated, an appropriately configured generator class is invoked in order to generate appropriate code from the data stored in the object model (Davidov, paragraph [0013]).

Claims 9 and 10 depend from independent claim 1, and claims 24 and 25 depend from independent claim 16. Applicants respectfully submit that Cheng and Davidov, alone or in any reasonable combination, do not disclose or suggest all of the features of claims 9, 10, 24 and 25.

Applicants respectfully submit that Cheng and Davidov, alone or in any reasonable combination, fail to disclose or suggest at least the following feature of claims 9, 10, 24 and 25: "providing a user interface with a plurality of selectable parameters for a custom storage class, said custom storage class specifying the manner in which an automatic code generator creates source code corresponding to data referenced by said graphical model in said graphical modeling and execution environment."

As discussed previously, Cheng fails to disclose or suggest the above feature of claims 1 and 16 from which claims 9-10 and 24-25, respectively, depend. The teachings of Davidov do not supplement Cheng in such a way as to cure Cheng's failure to disclose or suggest the above feature of claims 9, 10, 24 and 25.

For example, Davidov fails to disclose or suggest a custom storage class, as required by claims 9, 10, 24 and 25. Davidov discusses generating source code corresponding to *an object model hierarchy including screens, forms and servlets*. In contrast, claims 9, 10, 24 and 25 all require generating source code corresponding to *data referenced by a graphical model*. An object model hierarchy as discussed in Davidov is not synonymous with a graphical model as recited in claims 9, 10, 24 and 25. Thus, Davidov also fails to disclose or suggest generating source code corresponding to data referenced by a graphical model, as recited in claims 9, 10, 24 and 25.

Application No.: 10/698,820

Docket No.: MWS-062

For at least the reasons presented above, Cheng and Davidov, alone or in any reasonable combination, fail to disclose or suggest the features of claims 9, 10, 24 and 25. Therefore, Cheng and Davidov do not support a valid 35 U.S.C. §103(a) rejection of claims 9, 10, 24 and 25.

C. Claims 11 and 26

The Examiner rejected claims 11 and 26 under 35 U.S.C. 103(a) as being unpatentable over Cheng in view of DeMaster (office action, paragraph 16). Applicants respectfully traverse the 35 U.S.C. §103(a) rejections of claims 11 and 26 as set forth below.

The Cheng reference has been summarized above.

The DeMaster reference discusses a Java native interface code generator to facilitate mixed language programming (DeMaster, column 2, lines 5-17). The Java native interface code generator makes native code programmed in a native language, such as C, C++ or Assembly, accessible to Java application programs (DeMaster, column 2, lines 5-17). A programmer initially generates a native interface definition for native code, preferably using a neutral Interface Definition Language (IDL) to describe native code components, namely, the functions, data structures, constants and other user-defined data types, for which native mappings are to be generated (DeMaster, column 2, lines 5-17). The Java native interface code generator also supports complex data types, including nested data structures and multi-dimensional arrays (DeMaster, column 2, lines 5-17).

Claim 11 depends from independent claim 1, and claim 26 depends from independent claim 16. Applicants respectfully submit that Cheng and DeMaster, alone or in any reasonable combination, fail to disclose or suggest all of the features of claims 11 and 26.

Applicants respectfully submit that Cheng and DeMaster, alone or in any reasonable combination, fail to disclose or suggest at least the following feature of claims 11 and 26: "providing a user interface with a plurality of selectable parameters for a custom storage class, said custom storage class specifying the manner in which an automatic code generator creates

Application No.: 10/698,820

Docket No.: MWS-062

source code corresponding to data referenced by said graphical model in said graphical modeling and execution environment."

As discussed previously, Cheng fails to disclose or suggest the above feature of claims 1 and 16 from which claims 11 and 26, respectively, depend. The teachings of DeMaster do not supplement Cheng in such a way as to cure Cheng's failure to recite the above feature of claims 11 and 26.

For example, DeMaster fails to disclose or suggest a custom storage class, as required by claims 11 and 26. DeMaster discusses generating code corresponding to *user-derived native interface definition*. In contrast, claims 11 and 26 both require generating source code corresponding to *data referenced by a graphical model*. A user-defined native interface definition as discussed in DeMaster is not synonymous with a graphical model as recited in claims 11 and 26. Thus, DeMaster also fails to disclose or suggest generating source code corresponding to data referenced by a graphical model, as recited in claims 11 and 26.

For at least the reasons presented above, Cheng and DeMaster, alone or in any reasonable combination, fail to disclose or suggest the features of claims 11 and 26. Therefore Cheng and DeMaster do not support a valid 35 U.S.C. §103(a) rejection of claims 11 and 26.

Application No.: 10/698,820

MAY 22 2007
Docket No.: MWS-062**CONCLUSION**

In view of the foregoing amendments and arguments, Applicants believe that all claims should be passed to issuance. Should the Examiner feel that a teleconference would expedite the prosecution of this application, the Examiner is urged to contact the Applicants' attorney at (617) 227-7400.

Please charge any shortage or credit any overpayment of fees to our Deposit Account No. 12-0080, under Order No. MWS-062. In the event that a petition for an extension of time is required to be submitted herewith, and the requisite petition does not accompany this response, the undersigned hereby petitions under 37 C.F.R. §1.136(a) for an extension of time for as many months as are required to render this submission timely. Any fee due is authorized to be charged to the aforementioned Deposit Account.

Dated: May 22, 2007

Respectfully submitted,

By Kevin J. Canning
Kevin J. Canning
Registration No.: 35,470
LAHIVE & COCKFIELD, LLP
One Post Office Square
Boston, Massachusetts 02109-2127
(617) 227-7400
(617) 742-4214 (Fax)
Attorney/Agent For Applicant

Attachments